



LAUREA
AMMATTIKORKEAKOULU

Uuden edellä

The Design and Development of a Mobile Application Using SOAP Web Services for Increasing Business Mobility: a Case Study of Evolvit Oy

Mihaleva, Elena

2015 Laurea Leppävaara

Laurea University of Applied Sciences
Laurea Leppävaara

The Design and Development of a Mobile Application Using SOAP Web Services for Increasing Business Mobility: a Case Study of Evolvit Oy

Elena Mihaleva
Business Information Technology
Bachelor's Thesis
October, 2015

Mihaleva, Elena

The Design and Development of a Mobile Application Using SOAP Web Services for Increasing Business Mobility: a Case Study of Evolvit Oy

Year	2015	Pages	49
------	------	-------	----

The continuous growth in the use of mobile services, Smart Mobile devices and web services creates opportunities for building applications for people to use despite their location. This thesis examines the whole process of the design and creation of a mobile application for Evolvit Oy using web services, SOAP, WSDL and C#.

The Hour Tracking mobile application created for Evolvit Oy is a web service based application for improving accessibility to their existing services. The mobile application enables its users to access their own web service environment and record their favourite tasks and working hours from anywhere using a Windows Phone.

The customer requirements were identified and different technological and structural solutions were discovered from the student to design and create a user-friendly web service interface. The software development life cycle and agile methods were applied during the development process to assure the excellent quality of the development. A set of design methods were applied when designing the hour tracking application for the creation of a user-friendly customer interface.

The mobile application was developed on Visual Studio for windows phone 8.1 to test and prove the usability of the Web services. The hour tracking mobile application is created to provide Evolvit Oy's employees access to their web environment data, using the XML and WSD documents and SOAP web services.

Key words Mobile development, SOAP, web services, WSDL, mobility

Mihaleva, Elena

Mobiilisovelluksen suunnittelu ja kehittäminen käyttäen SOAP www-sovelluspalvelua liiketoiminnan liikkuvuuden parantamiseksi: tutkimus Evolvit Oy.

Vuosi	2015	Sivumäärä	49
-------	------	-----------	----

Mobiilipalveluiden, älylaitteiden ja internet palveluiden jatkuva kasvaminen antaa uuden mahdollisuuden sovellusten luomiseen joita käyttäjät voivat käyttää sijainnistaan huolimatta. Tämä opinnäytetyö kattaa koko mobiilisovelluksen suunnittelun ja tekemisen prosessin www-sovelluspalvelua sekä SOAP, WSDL ja C# -kieliä hyväksi käyttäen.

Evolvit Oy:lle luotu internet palveluun perustuvan Hour Tracking mobiilisovelluksen on tarkoitus parantaa olemassa olevaa liiketoimintaa. Mobiilisovelluksen avulla käyttäjät saavat mahdollisuuden päästä käsiksi yrityksen omaan internet ympäristöön ja tallentaa heidän suosikki tehtäviään sekä työtunteja sijainnista riippumatta käyttäen Windows Phone alustaa.

Asiakkaan vaatimukset saavutettiin ja erilaiset teknologiset ja rakenteelliset ratkaisut löydettiin jotta voidaan suunnitella ja luoda käyttäjäystävällinen internet palvelu. Ohjelmiston kehittäminen elinkaarta ja Agile menetelmiä hyödynnettiin, jotta voidaan varmistaa kehityksen erinomainen laatu. Joukko suunnittelu menetelmiä otettiin käyttöön suunnitellessa Hour Tracking sovelluksen käyttäjäystävällistä käyttöliittymää.

Mobiilisovellus suunniteltiin käyttäen Visual Studiota Windows Phone 8.1 -käyttöjärjestelmälle jolla testattiin ja todistettiin internet palveluiden toimivuus. Hour Tracking sovellus on luotu käyttäen XML ja WSDL tiedostoja sekä SOAP www-sovelluspalvelua jotta voidaan antaa Evolvit Oy:n työntekijöille pääsy heidän omaan verkko ympäristönsä.

Avainsanat: Mobile kehitys, SOAP, www-verkkopalvelut, WSDL, liikkuvuus

Table of contents

1	Introduction	6
1.1	Company background	7
1.2	Study Benefits.....	8
1.3	Objectives	8
1.4	Limitations	8
1.5	Terminology	9
2	Methodology	10
2.1	Research approach	10
2.2	Software Development Life Cycle	10
2.3	Agile Development Method	11
2.4	Research Methods	12
2.4.1	Interview	12
2.4.2	Benchmarking	13
2.4.3	Wireframe	13
2.4.4	Prototype	13
3	Theoretical background and knowledge base	14
3.1	Overview of web service technology	14
3.2	Web service architecture	15
3.3	Understanding XML	16
3.4	Understanding SOAP	18
3.5	Understanding WSDL	19
3.6	Understanding C# and Visual Studio	20
3.6.1	C# Basics	20
3.6.2	Microsoft Visual Studio.....	21
4	Hour tracking application development	22
4.1	Company requirements.....	22
4.2	Choosing the development method.....	23
4.3	Hour tracking application design.....	23
4.4	Application structure and development	26
4.5	Testing	29
4.6	Implementing/ publishing and maintenance.....	31
5	Evaluation	31
6	Conclusion.....	32
	Appendices.....	37
	Appendix 1: Customer Requirement Questionnaires	38
	Appendix 2: Snapshot of Company Requirement.....	39
	Appendix 3: Customer Feedback	40
	Appendix 4: Prototype	41

1 Introduction

The widespread use of mobile devices is an important consideration for any business. According to eMarketer's article, around two billion people worldwide will be smartphone users by 2016. People use smartphones, tablets and mobile devices on a daily basis. Mobility has become part of the present-day as well as part of the business life (eMarketer 2014).

Businesses are going in a general direction of implementing mobility within the business services and employee's way of working. Different campaigns are used for implementing mobility such as: wireless technology and bring your own device (BYOD). BYOD is a policy that allows employees to bring their personal devices (laptops, smart phones, tablets) to tier work place. Those devices are generally used for accessing private business information and applications. According to CPS strategy's article "Growing Importance of Mobile Devices in Business [Infographic] ", "96% of businesses in the USA use wireless technology". The main reasons for companies to use mobile applications are growing profitability , preserving time and developing productivity (Kerner 2014).

Implementing a mobile strategy into the business life is proven to help the business. Mobility is the main drive of IT transmutation in the present days. It changes how people perform their work and leads the organizations to success (citrix.com 2013).

The biggest benefit from mobility for the employees and the business is the capability of performing the tasks faster. 55% find greater flexibility to meet customer's needs. According to the findings in the "Mobility in Business report" made by citrix.com 47% of the survey responses state that the mobility improves motivation and productivity of the employees. In Error! Reference source not found.

This thesis examines a software solution developed for Evolvit Oy to facilitate mobility into the company and later on in the business users. Special attention will be addressed to the design, development and implementation of a customized mobile application into the business, as a primer writing and research topic of this project.

1.1 Company background

Evolvit Oy is a small company providing IT solutions. The company was founded in 1991 and from then has been providing their costumers with software and services. The main areas in which Evolvit is concentrating are Business intelligence and e-business services. The most known products of the company are HRDesk and CRM Service (Evolvit Oy No Date).

The main office of Evolvit is located in Espoo, but the company has other two offices located in Tampere and Jyväskylä. At the moment the company staff is approximately 60 people. The company is aiming to expand in the nearest future.

Evolvit Oy had an outdated hour tracking system. The company wanted to improve and modernize their system. The company needed the system to be modern and to allow mobility for tracking and recording the working time of their employees.

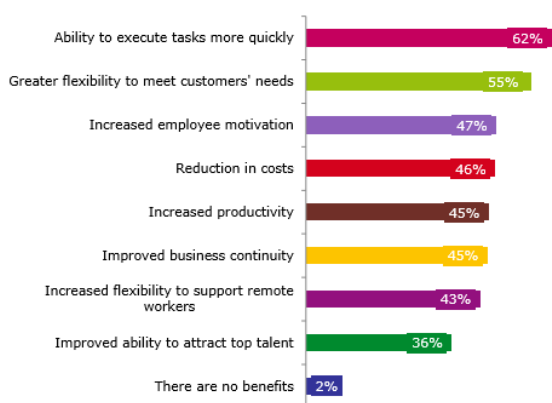


Figure 1 The business benefits of being a mobile organization

https://www.citrix.com/content/dam/citrix/en_us/documents/news/citrix-mobility-in-business-report.pdf

The company requires a mobile application that track employees' working hours, and in addition record their work related traveling kilometres. The application needs to save the tracking information to the newly created system, and it should be platform dependent (made for Windows phone only).

1.2 Study Benefits

The main purpose of this study is to develop an hour tracking application for Evolvit Oy using a web service. Understanding the theory of web services and the conception of mobile development will help creating a well functioning and easy to use application. The main question is how the mobile application will increase the mobility in the company and bring benefits to the company.

First of all, the mobile application is suitable on any type of windows phone devices. Each of the employees working in Evolvit Oy have a Windows Phone as a work mobile device. That makes the recording of their work hours more convenient and time saving. Also it increases the flexibility of remote recording of their work. Because of the recording of the work time becomes easier it also improves the work satisfaction.

1.3 Objectives

The main objectives of this study are to design a user-friendly and developing well functioning mobile application by using SOAP web services to insert mobility into the hour recording system of the company.

The main features of the application to be developed are a function for entering the working hours and a function for editing existing work tasks. Also a feature for adding task to a favorite list of tasks and option to edit the list.

The advantages for the company of having a good hour tracking system are better time management, focus and workflow. As additional benefits for the company would be more accurate salary paying and client billing.

1.4 Limitations

During the second requirement meeting held on February 18, 2015, the developer had difficulties to fully understand what the company representative wanted exactly. The developer was not familiar with the meaning of the abbreviations “SOAP” and “WSDL”. Which were the main languages that needed to be the implementation of the mobile application.

During that meeting the development requirements were changed and the development process was limited as following:

- Only limited to Windows phone development;
- Limited to use only C# and XAML as developing languages;
- Limited to use SOAP web services through WSDL file;

Another limitations during the study process were the outdated literature on the requested languages: SOAP and WSDL. Not only the literature was outdated but there were almost no sources available to read how to implement the languages.

1.5 Terminology

In this section the most important terminology used in the study will be explained. It is important to have knowledge of the terms used during the thesis process in order to understand the overall idea and the findings of the case study. Business Mobility or Enterprise Mobility is the term used for expressing the change of the work habit of the employees towards working remotely out of the office and using devices which allow mobility in the work task accomplishment (Rouse No Date).

Mobile Application is a software created to be used on mobile devices. This kind of programs are specially designed to be executed on smart phones or tablets (Wikipedia 2015). Mobile applications are used in businesses to connect with their services to increase its mobility of the operations. The method used for connecting a internet based service with mobile application on a device is called Web Services (Wikipedia 2015). This is the method which allows the mobile applications to communicate to eachother and exchange data.

The communication between application over internet cannot happen without protocols and web languages. SOAP (Simple Object Access Protocol) is a protocol , which allows exchanging information in specific structure over the network (Wikipedia 2015). WSDL is a file, which describes the end points of the communication and contains of messages with information. WSDL files have been designed to test if the SOAP-based web services are working (Web Services Description Language (WSDL) 1.1 2001).

The last important part of the communication between web service and a mobile application is the Cookie. A cookie is a function in the mobile application, which assist the requests made between the Web Service and the browser. The cookie remembers the users information and the Web Service can read it. In this case the web service knows exactly which information to send back to the mobile application (Microsoft 2015).

2 Methodology

This study implements qualitative methods in the research stage of the hour tracking application development. The applied methods in this case study are interview, analysis, and benchmarking. The results from the qualitative methods have great impact on development of well designed and functioning mobile application.

In the following subsections are described the methods used during the research stage of the thesis, and the design and development of the mobile application.

2.1 Research approach

The research approach that this development case study applies is constructive research. The main objective of the constructive research is to recognize and find solutions to real functional issues. "The identified research problems are used to propose research questions that address the problem. The questions are solved by developing or constructing a solution which will be operationalized to determine its workability and appropriateness." (Adekunle 2011).

Therefore, the constructive research is the most acceptable choice for a development of a real product, a mobile application. The gathered outcomes from the constructive research methods will be evaluated and used in the development of the hour tracking application.

2.2 Software Development Life Cycle

The process of structuring and building the mobile application is based on the Software Development Life Cycle (SDLC). SDLC is the process used in the software industry for assuring design and development of excellent quality software (Justin 2013).

In Figure 2: Software Development Life Cycle is displayed the five stages of SDL:

- Planning, defining and analyzing the user requirements;
- Designing the product;
- Developing the product by the created design;
- Test the product for issues;
- Maintain the product;

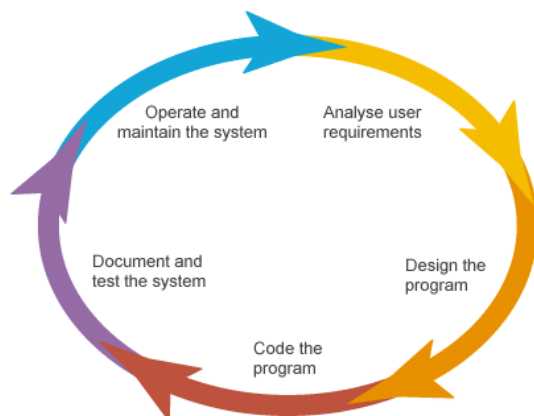


Figure 2: Software Development Life Cycle

https://airbrake.io/blog/wp-content/uploads/2013/07/development_life_cycle.gif

There are various SDLC models, each follows unique pattern of steps, in order to ensure excellent accomplishment in the process of software development.

The most common SDLC development models are Waterfall, Iterative, Spiral, and Agile Models. For this particular project the most suitable of the various models is the Agile SDLC. The reasons for choosing the Agile SDLC are described in the following section.

2.3 Agile Development Method

The SDLC Model chosen for this case study is the Agile Model. Agile SDLC focuses on quick delivery of functioning software product by adaptable process and constant user feedback. In Figure 3 Agile Model is illustrating the series of steps during the agile development process. The Agile Model was the most suitable model because it delivers functioning software fast, it receives user feedback constantly and it gives freedom for the development.

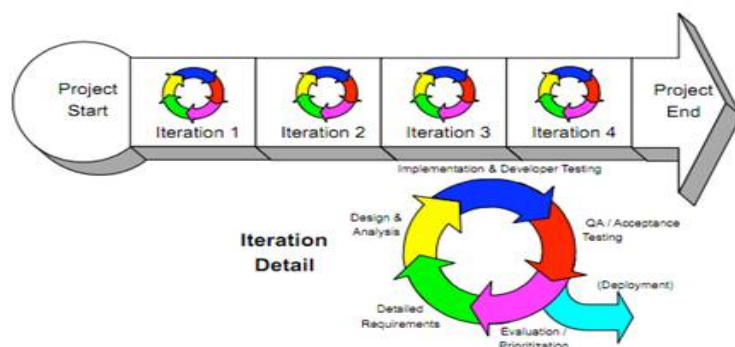


Figure 3 Agile Model

<http://scrumreferencecard.com/scrumreferencecard-content/uploads/2013/03/scrum-iteration-detail.png>

The Agile Model separates the whole project into smaller parts called iterations. Each iteration is lasting between one to three weeks and includes the steps of planning, analysing, designing, building, and testing. After each iteration the outcome is presented to the user and feedback is received. If the user satisfaction is achieved the product is released into the market, if not the process continues with the next iteration.

2.4 Research Methods

Analysis of mobile design and development books, websites, articles and tutorials were used to find the best practices in the mobile development and design.

When designing and developing a software or any other service, the designer should aim for innovation. Innovation is the value provider of the service. It is important to be innovative, to bring a new value to specific area and time, but innovation failure rates are still high (Innovation failure rate statistics).

According to Kumar, there are four main design principles for an innovative service to be successful. The most important principle is called “Building Innovations around Experiences”, which is accosted with the term “User Experience” (UX). The user experience is the key factor which will make your product successful. If more companies concentrate on understanding their customers, that will lead them to great innovation success. (Kumar 2012).

Wireframe and benchmarking design methods were used in this study, to assure successful innovation and good user experience of the mobile application. Interviews and benchmarking were conducted for further support to the findings of the best practices. The following subsections describe in details the design methods used in the study.

2.4.1 Interview

Interview is a research method for providing insight information on daily life experience, people behavior, thinking and reacting. The interviews provide with data for analysis. This data can be in written, video or voice recording format. The analyzed data is used as support and guideline for the development process. Often the best solution come from the interviews. (Stickdorn 2012)

In this study the type of interview used is structured interviews. Structured interviews are used for quantitative research and they gather data for analysis. During the study two interviews were conducted. The first interview was developed on 11.02.2015 in Laurea Leppävaara campus. The interview aimed to gather the company requirements, development wishes and future maintenance requirements.

The second interview was contributed to gather data for customer satisfaction, development and design feedback and evaluation of the application. The full interview questions and answers are available in the following Appendix 1: Customer Requirement Questionnaires.

2.4.2 Benchmarking

Benchmarking is the action of examine in contrast existing products, their performance and best practices. During the process the best companies in the required field are identified and compare their products to the own product. Benchmarking is used method for making improvements on the developing service, by comparing it to the best similar services provided on the market. The main objective of the method is to learn from these best in the field, already created businesses and add or improve features in the service, which will be created. (Stroud no date).

Benchmarking was contributed to examine similar mobile applications products in contrast. The area in which the comparison was aiming are as following: the mobile application visual appearing and the mobile application features. The outcome of the benchmarking was a list of the top six best hour tracking applications and list of their features.

The list was used to compare if the best practice includes the same features as the features required from the case company Evolvit Oy.

2.4.3 Wireframe

The wireframe is used for graphically illustrate the visual layout and the functions of the mobile application. It is also used for representing the vision of the product to the company and receiving feedback on the design and functional ideas.

In this case study the wireframe was established to demonstrate the ideas for design and functionalities of the mobile application in early stage of the project. The wireframe was presented to Evolvit Oy for receiving a feedback, suggestions, or change requests.

2.4.4 Prototype

The prototype is a mock-up example, which development aims to represent the visual representation and behaviour of the end product. The prototype allows you to create interactive wireframe, and visualize the design better before development of the product (Banfield 2009).

In this case study the prototype was developed for better visual re-creation of the mobile application. The prototype allowed to test the design and see it on a real mobile device. The prototype represented the wireframe to look and feel like a real mobile application, which provided more accurate feedback from the company Evolvit Oy.

3 Theoretical background and knowledge base

To have better understanding of how the mobile applications work with SOAP and WSDL, one must have better knowledge of what is web services and how they work. The following subsections will introduce the web services, SOAP and other technologies as Web Service Description Language (WSDL) and XML.

3.1 Overview of web service technology

Web service is a technology used for implementation of service-oriented architecture. The purpose of this technology is providing of interoperation between different programming languages, environments and platforms.

Wikipedia explains the web services as a mechanism for communication between two electronic devices over internet (WIKI). Web services are used primarily by organizations, to utilize communication between the employees and with the clients. Web services enable businesses to exchange data without intensely understanding of each other's IT systems.

The standard-based user interface is abstract, which allows the service application and the mobile application to be written in different programming languages. Web services can be interoperable, the web services and mobile devices can exchange and interpret that shared data, which makes the platforms insignificant (Snell, Tidwell & Kulchenko 2001).

In the following subsequent chapters are covered in details these topics.

- XML (extensible Markup Language): a base language for all standards over internet protocol foundation. It is a markup language used for labeling the data and the data structure.
- SOAP (Simple Object Access Protocol): a communication protocol used to assign the data and allow the end-points to communicate with each other.
- WSDL (Web Service Description Language): a standardized mechanism used for specifying the Web service. The WSDL document describe the operations, parameters, data types a Web service provides.

3.2 Web service architecture

Web service architecture is a new model for software architecture. It consist of the communication between three roles: service provider, service registry and service consumer. The interaction includes publish, find and bind operations, as shown in Figure 2 Web service architecture.

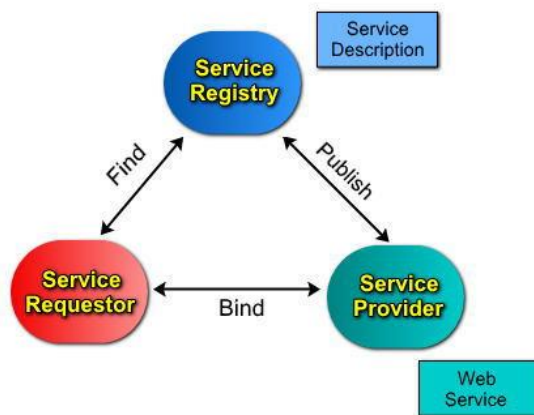


Figure 2 Web service architecture

<http://www.ibm.com/developerworks/library/ws-wslover/WebServicesArchitecture.jpg>

In the Web service architecture the service roles:

- **Service Provider:** the owner of the Web service. The web service is created and the service provider published a description of the service and it's access information to the service registry.
- **Service Requestor:** Also known as service consumer. Service requester can be a person using browser or another program. The service requester looks into the service registry in order to find a match that meets the service requirements. The service consumer can be more than one in the service architecture.
- **Service Registry:** Also is known as service broker. It creates the interface for the web service and allow access information to any service requester.

The Web service operations performed by the service roles as following:

- **Publish** is the operation assigned to the service provider. It needs to publish the description of the service registry. Publishing the description is optional.
- **Find** is the operation assigned to the service consumer. It finds the service description directly, or search for the location of the required service in the service registry.
- **Bind** is the process, in which the service consumer invoke the service using the data provided by the service registry.

Every web service has its own custom communication protocol through which the service consumers can access the service provider. The most frequent occurring model for accessing the web service is requests and responses (Figure 5 Web service accessing model). The client send a message to the service, which describes the operation that needs to be performed. The service provider perform the described operation and sends a response message to the service consumer (Jenkov, 2014).

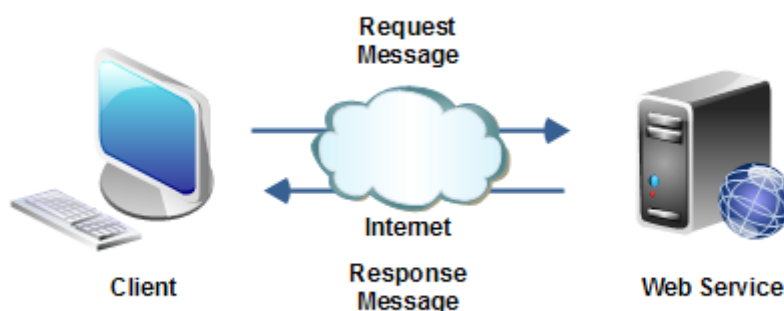


Figure 5 Web service accessing model

The web service has a custom user interface through which the service consumer access the service provider. Every custom interface needs to be standardized, according to World Wide Web Consortium (STANDARDS 2015). W3C is an international community, which provides a global recognizable protocols for discovering, designing and accessing the custom interface of the web service. There are XML- based standards which are recommended to use.

These standards consist of two important components: Simple Object Access protocol (SOAP) and Web Service Description Language (WSDL). The primary advantages of the Web Services is that they are easy to access and use. They are easy to access because of using the same technology as the web browsers: World Wide Web (WWW). They do not have any firewall issues like most of the programming languages.

3.3 Understanding XML

According to World Wide Web Consortium (W3C 2015), External Markup Language (XML) is a tractable and simple text-based language formed from Standard Generalized Markup Language (SGML ISO 8879). XML was first created to describe data and to carry information as a software- and hardware-independent instrument. XML is data oriented language, which concentrate on describing data and requesting additional information about data identity. It can be used not only to describe data but also to store, carry and exchange data.

XML can be seen as interoperable. It makes possible to exchange data between platforms, which have not been design to work together. XML is hardware and software independent instrument for communication between incompatible systems.

XML Document Example

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to> Maria</to>
  <from>Elena</from>
  <heading>Requesting</heading>
  <body>Don't forget to bring me back my umbrella!</body>
</note>
```

XML document consist of elements each starting with start tag “<” and end with closing tag “>”. The first line - the declaration - describe the version and the character encoding of the XML document. The following line define the root element of the document. In this case the root element is “<note>” indication that this XML document is a note. The root element has four child elements: to, from, heading and body. The child elements give details on from whom the note is, what is the text in the note and to whom the note is. The last line of the XML document is the end of the root element.

The XML document described above can give a brief understanding of what the XML document looks like. XML syntax has predefine rules, which can be found in W3CXML 1.0 (Extensible Markup Language (XML) 1.0 (Fifth Edition), 2013). Some of the rules are:

- XML tags are not predefined.
- The root element can be only one.
- All XML element should have opening and closing tags.
- XML tags are case sensitive.
- Cannot overlap XML elements.

If two different element used the same tag name, it will cause an error. That is due to the fact that the XML element names are not fixed. The solution for that collision is assigning a namespace with a unique name to each tag.

3.4 Understanding SOAP

In the previous subsections was explained that Web Services is a communication between devices and XML is the basic language for that communication. Like every language XML need a grammar. Simple Object Access Protocol is the grammar, which gives authority to the systems to talk to each other.

According to Gudgin , SOAP is an agile protocol designed for delivering structured data over less concentrated environment. It operates using XML technology to establish an extensible messaging scheme for message exchange over a range of fundamental protocols. The framework has been constructed to be self-relevant and independent of other programming frameworks.(Gudgin 2007).

The most important point to remember from this definition are:

- SOAP is XML based technology.
- SOAP can use variety of network protocols for transmitting its messages.
- SOAP is interoperable: platform, language and environment independent.

SOAP messages are ordinary XML documents which includes SOAP Envelope, SOAP Header, SOAP Body, and SOAP Fault. The structure of the SOAP message is showed in Figure 6 SOAP Message Structure.

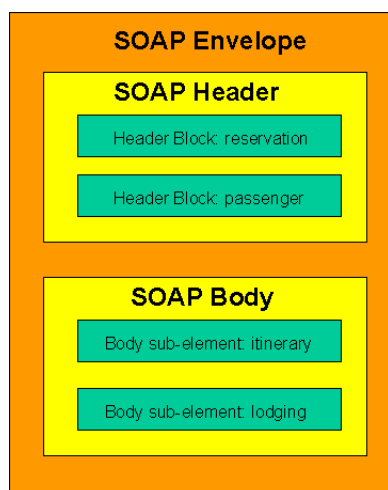


Figure 6 SOAP Message Structure

<http://www.w3.org/TR/soap12-part0/primer-figure-1.png>

- **SOAP Envelope** - It is mandatory part of the SOAP Message structure, which defines the beginning and the ending of the message.
- **SOAP Header** - It is optional part of the SOAP Message, which could include optional attributes for processing the message.
- **SOAP Body** - It is necessary element of the SOAP Message, which includes the XML data with the message being sent.
- **SOAP Fault** - It is an optional element, which could include information for the errors that appear while processing the message.

SOAP messaging framework is independent, which means it can be sent over several different transport protocols like HTTP, SMTP, FTP, etc. The most command used transport protocol is HTTP, because HTTP is supported by most of the internet browsers.

3.5 Understanding WSDL

Web Service Description Language (WSDL) is a language for describing and locating network services and the way of accessing them. The WSDL document is document written in XML format, which contains a group of descriptions of the fundamentals of the web services.

The main elements for describing a web service used in WSDL documents are type, message and binding. In the following table each of the elements used in WSDL documents are described.

Elements	Description
<type>	A container for data type definitions used by the web service
<message>	A definition of the type of the communicated data
<portType>	A collection of activities supported by the endpoint/s
<bindings>	A specification of the protocol and data format specification for <portType>

Table 1 WSDL Elements

The <portType> is the most important of the WSDL file. It describes the operations that can be performed and the messages that can be involved. The <message> defines the data elements of an operation. The <type> defines the data types that are used by the web service. The <binding> defines the data format and protocol for each <portType>.

3.6 Understanding C# and Visual Studio

3.6.1 C# Basics

C# is a programming language developed by Microsoft Corporation 15 years ago. According to Hejlsberg, “C# (pronounces “See Sharp”) is a simple, modern, object-oriented, and type-safe programming language” (Hejlsberg 2011). The most important statements to remember from this definition are:

- C# is object-oriented language. In other words the programming model of the language is based on objects. Each object structures data containers (attributes) and code of different functions (methods). The objects within the program can interact with one another and can access and modify the attribute’s data.
- C# is type-safe language. Type-safe code stands for code means the code has the authority to access only well-defined and allowed to access values.

In the following lines is shown a simple application code written in C#. The example:

```
using System;
public class Hello2
{
    public static void Main()
    {
        Console.WriteLine("Hello, World!");
    }
}
```

In the example are shown a class and two methods. The class is public, which means that it is accessible by all. The name of the class is Hello2, which follows that the name of the file in which the class is stored is Hello2.cs. Every class should contain a Main method within. The second method is called WriteLine method, which is used to display the string “Hello, World!” as output.

The output of the program will be: Hello, World!

3.6.2 Microsoft Visual Studio

The main development tool for this study is Microsoft Visual Studio (Visual Studio). The reason of the choice for this tool is because it met all the requirements needed for the development of the hour tracking mobile application.

Here are some of the main advantages that Visual Studio has:

- It is a software application developed by Microsoft. Visual Studio is an integrated development environment (IDE), or in other words computer program used from software developers for creation of variety of software for Microsoft Windows, websites, web and mobile applications and web services.
- It can produce both native and managed code for native or hybrid application development.
- Visual Studio as any other IDE consist of editor for source code, and build-in tools and a debugger (Figure 7 Visual Studio Community 2013).
- The integrated debugger is the tool that is used for testing and debugging the source code as well as the machine-level code. Some of the build automation tools are forms designer, web designer, database schema and class designer.
- Visual studio also has the ability to include different kind of plug-ins for additional support and source control or adding new toolkits for visual design.
- Visual studio supports wide variety of programming languages (like Ruby, XML, HTML, JavaScript, CSS) and include the main built-in languages C, C++, VB: NET, C# and F#.
- Microsoft provides the free edition of its Visual Studio.

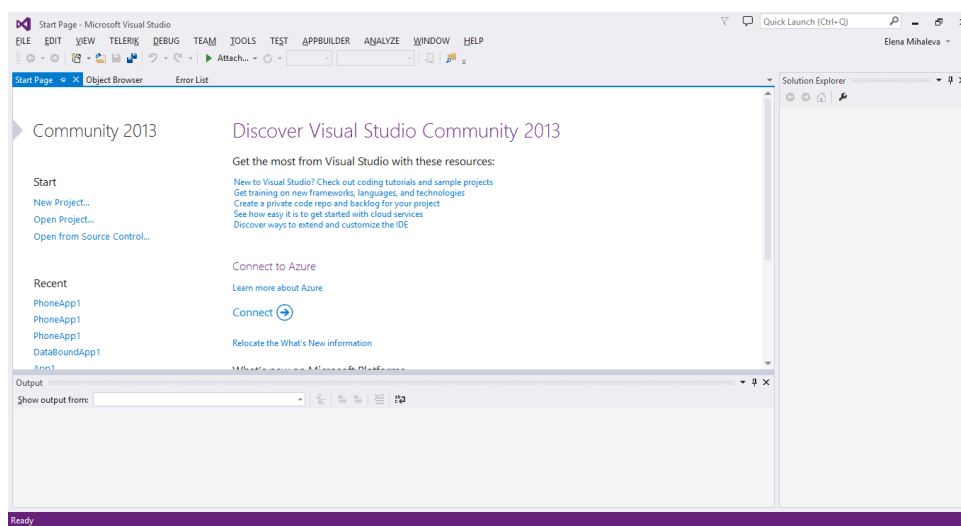


Figure 7 Visual Studio Community 2013

The combination of all these positive attributes brought up the selection of Microsoft Visual Studio as main development tool.

4 Hour tracking application development

The following sections will describe the design and implementation of the hour tracking mobile application, developed for Evolvit Oy. The development stages using the Agile Development Method will be explained in detail. The design and development of the mobile application are based on the research findings, the best practices, and the company requirements that you can see in the following section.

The development of the Hour tracking application started in January. A representative from Evolvit Oy arranged a meeting on January 18, 2015. The requirement for the development of the application were given on that same meeting. The development requirements are described in the following section.

4.1 Company requirements

In January 2015 was the first meeting with Evolvit Oy. The company contacted the student through the Senior Lecturer of the BIT degree. On the first meeting the student received the information regarding the project and the company requirements regarding the technology, programming and general skills.

The requirements for the development of the Hour tracking application were sent by email. The first requirements were as following:

- Mobile application suitable for all operating systems (Microsoft, Android and iOS);
- Creating a mobile application using HTML5 and phone gap
- User-friendly user interface;

On February 18, 2015 Evolvit Oy changed the requirements for their application. The new requirements were established during a meeting with the company representative. The company changed the requirement for the operating system of the mobile devices. They wished the development of the mobile application to be made only for Windows Phones. The second change that the company requested was the development language to be changed to C# and SOAP using WSDL file.

The company assigned the design of the mobile application to the student with full freedom from any requirements. The chosen design and the development of the hour tracking application are displayed in the following section 4.3 Hour tracking application design and 4.4 Application structure and development. More details about the company's requirements can be found in the Appendix 2: Snapshot of Company Requirement.

4.2 Choosing the development method

The choice of using Web Services as the main development method of this mobile application was established because of the following reasons:

- security reasons;
- platform and language independency;
- better performance;

First reason for choosing the Web Services is the security. There is secure access to the database and only the server and the app user has access rights.

The second reason was the portability of the Web Services. The application users does not need to use the same mobile device, to access to the same operating system. The developer just needs to re-create a good access layer to each platform, but it doesn't need to create the whole web service for each different device.

That ability of allowing different programming languages to communicate between each other was the most important reason for choosing Web Services, because of the company requirements of using WSDL file and SOAP service and creating a windows mobile application using C#.

The WSDL file received from the company includes SOAP messages, which are XML based. XML language does not communicate directly with C#, which is the main the programming language for Windows Phone application. Therefore the chosen development platform was Microsoft Visual Studio. The development platform has in build Service Reference tool, which translates WSDL file to C# programming language.

4.3 Hour tracking application design

The biggest influence on the design of the Hour tracking application is the design methods that were conducted. The benchmarking predetermine the user interface model used in the application. Extended research was made in the area of hour tracking applications available in the market. After finding the top six time tracking applications, a deep comparison was conducted. All the customer requirements were taken under consideration and the most suitable application was chosen for base user interface example.

The chosen application after the benchmarking was TSheets (TSheets, 2015). TSheets was chosen because of the following reasons:

- It is a hour tracking application
- It saves the hours to already existing online system
- You can enter hours and description of the tasks
- It has list of the tasks
- GPS tracking system;

All of the above reasons are the first requirements received from Evolvit Oy on the first meeting.

In Figure 8 TSheets Mobile Application is displayed the chosen example application. The design and functionalities of TSheest were used for base for the design and building of the Hour tracking application for Evolvit Oy.



Figure 8 TSheets Mobile Application

All the gathered ideas and list of features were reflected in the wireframe and later on in the prototype created in the early stage of the development of the study.

The first conducted design method is the wireframe. The tool used for creation of the mockup is a free online tool called Cacao. That tool allows you to develop wide range of mockups and to export them in several different formats. The layout of the wireframe used for the Hour tracking application is a mobile phone in three screen views, exported as a picture. The wireframe of the mobile application design is displayed in the following Figure 9 Wireframe.

The first screen view, in Figure 9 Wireframe , shows the main page when the application is loaded on the mobile device. The front page is a login page with username and password input fields and a login button.

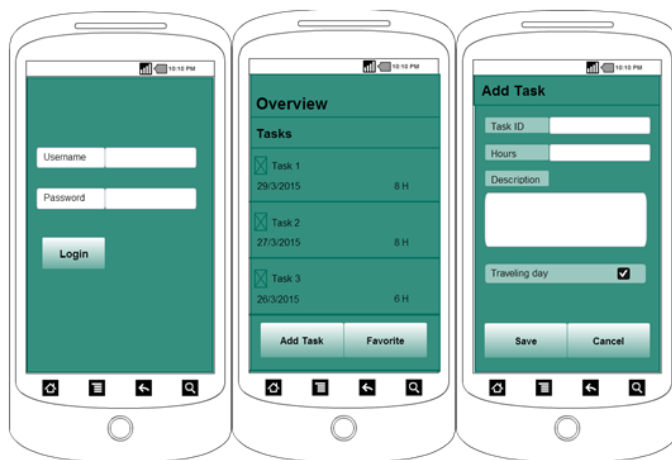


Figure 9 Wireframe

The second page displays a how the overview board should look like. The tasks and the working hours in a list and buttons for adding a task or for opening a favorite task. The last page shows how the “Add Task” page will look like when the user presses the “Add Task” button from the “Overview” page.

The wireframe was made simple and easy to understand. The main pages and ideas of how the application will be developed were displayed in the wireframe, in other hand the functionality of the application was shown in the prototype.

As mentioned in section 2.4.4 Prototype, the use of the prototype is to make the application real and functional before it is even created. The tool for making the prototype is called “Justin Mind”, which is a prototyping software with a 30 days of trial period. The tool was used because it created great applications and it is easy to use. The Hour tracking application’s prototype looked similar as the wireframe, with the difference that the buttons could be pressed and they opened a new page. The prototype can be seen in the Appendix 4: Prototype.

All the feedback from the design stage was gathered, analyzed and the outcomes were used as a base for the development of the hour tracking mobile application.

4.4 Application structure and development

The following stage of the SDLC (2.2 Software Development Life Cycle) after the Design stage is the Building phase.

The design of the visual and functional structure of the hour tracking application was used for a framework for the development process. In overall the hour tracking application was designed to have three pages: Login page, Favorites Page, and Add task page.

Each page would have been developed in its own iteration as shown in the following Figure 10 Development process.

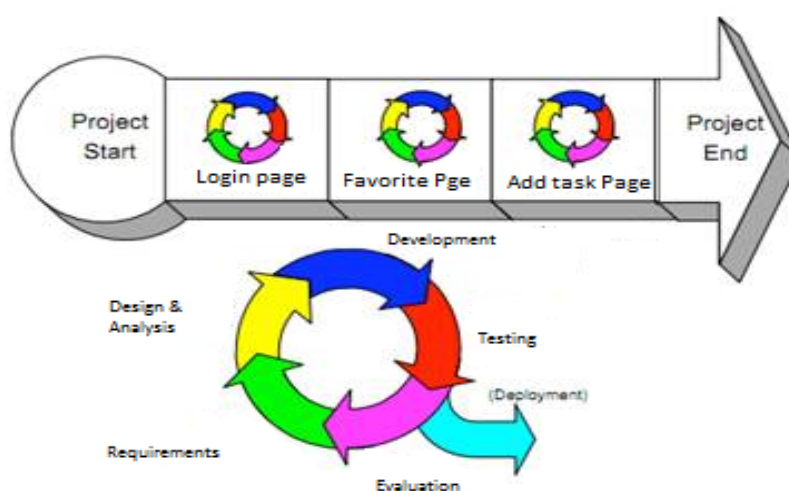


Figure 10 Development process

The development of the hour tracking application happened in three stages producing three versions of the application. The first version of the application included a default design and functional three pages. The second version had implemented the customized design and additional features were made. In the third version the last additional changes were made after the representative from Evolvit tested the application.

During the first phase the mobile application was created and structured. First the application was created by using the Microsoft Visual Studio editor. The application is a Windows Phone 8.1 application using C# for primary language. The three pages from the wireframe were created and the WSDL file was used for creating a Service Reference SOAP connection with the web service.

First version of the mobile application, as shown in Figure 3 Hour Tracking Application v1.0 had the default visual layout of the Windows Phone application.

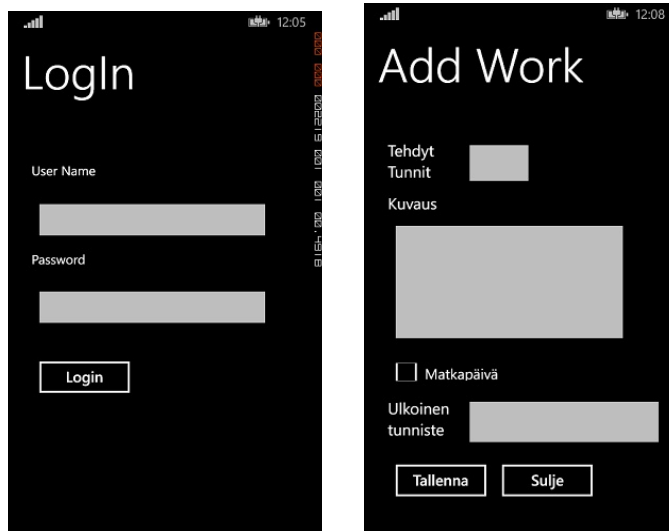


Figure 3 Hour Tracking Application v1.0

The second version of the mobile application was customized with the colors scheme of the case company, the layout was changed and the company logo was added. The customized login page after the change is shown in Figure 12 Hour Tracking Application v.2.0.

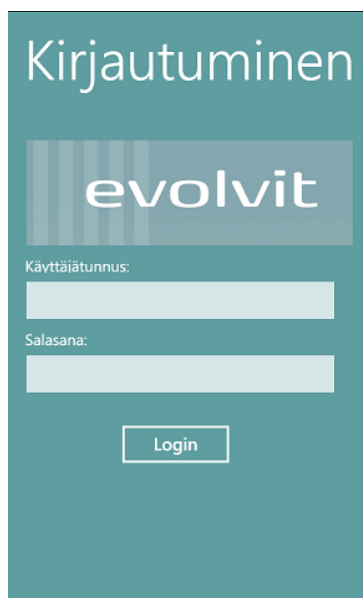


Figure 12 Hour Tracking Application v.2.0

The final version of the mobile application consisted of new features as:

- adding a traveling day;
- showing the name of the client, project and task; the task number;
- calendar;
- customized message boxes;

At the end the final hour tracking application was tested and evaluated if all the features are working and the layout looks good. The end product is displayed in the following Figure 13 Hour Tracking Application Final Version.

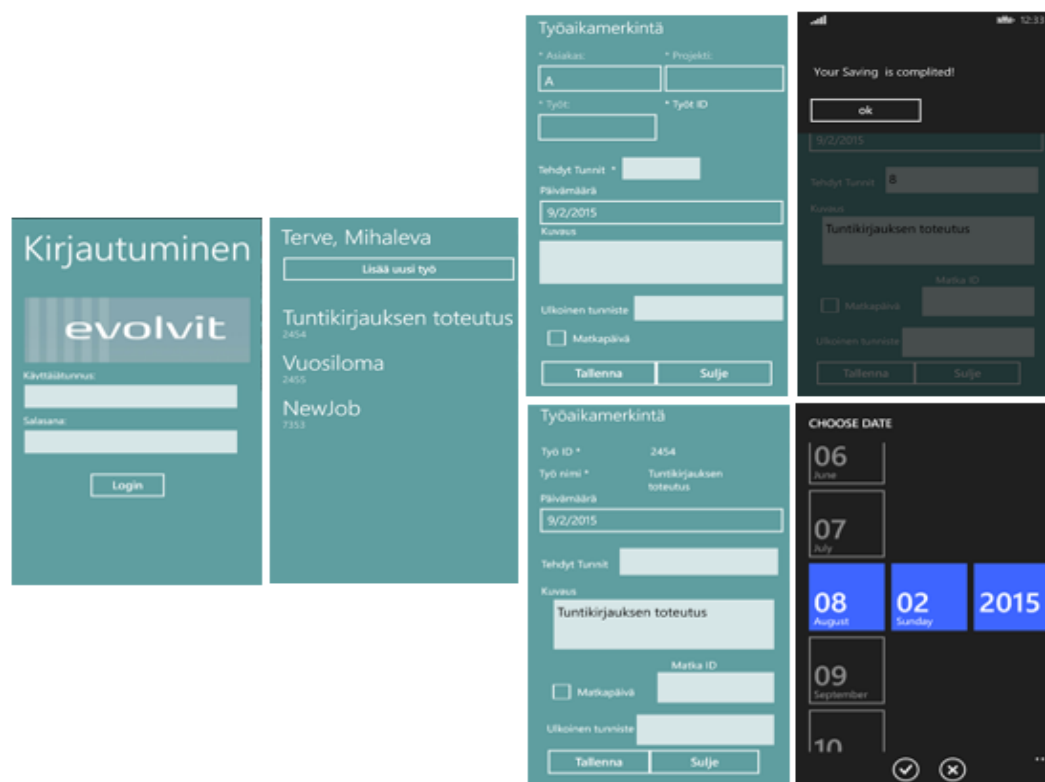


Figure 13 Hour Tracking Application Final Version

4.5 Testing

The fourth stage after the development according to SDLC, described in section **Error! Reference source not found. Error! Reference source not found.**, is the Testing. Because of using the Agile method of development the testing took place several times during the development of the hour tracking application.

Testing is made after every iteration of the development process. In other words every component of the mobile app was tested and evaluated after its development. The testing and improving of the product takes the most time of the development process.

As already mentioned in the previous section each page creation had its own iteration, which mean each page was independently created and tested before putted together. The first created and tested page was the “Login page”. This page make the connection between the mobile application and the web service. The first test failed with the following error message: “Server was unable to process request”, displayed in the Figure 14 First Testing Error Message.

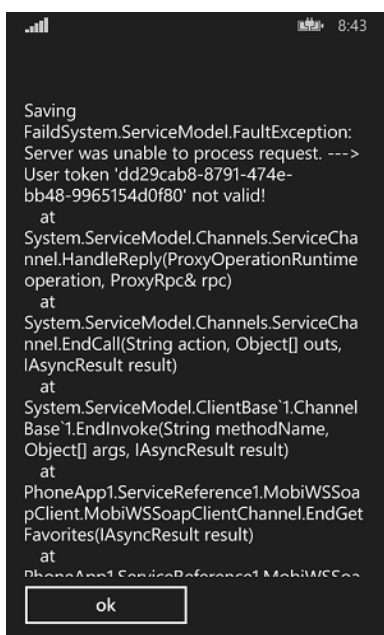


Figure 14 First Testing Error Message

Solving the issue took several weeks. The issue occurred because of missing Cookies class on the mobile application.

The line of code added in the “Login page” was:

```
App.soap.CookieContainer = new CookieContainer();
```

This one line of code is the most important part of the login page. Without the CookieContainer no connection could be established with the web services. The error was caused by mismatching communication between the application and the web services. When a user login with its own username and password that session is given a key. That key is unique for each login. In this case, before using the CookieContainer, the user was given a key, but there was no place to store it. When user was attempting to use other function, like retrieving its favourite task list, the application was giving a new key, which was mismatching with the one given with the login. After creating the CookieContainer, the error was solved, because the same unique key was used for each operation.

At last after implementing the function for the Cookies the connection between the web service and the phone browser was successfully established and the test was passed.

Even though the “Login page” took more than a month to be passed to the next stage of the SDLC, at the same time the other pages were also tested. Because of using the agile method of development the process can continue if one stage couldn't be passed.

The “Favourite's page” includes a list of the favourite tasks from the web service. The page was tested to see if after login it can retrieve the data from the web service and displayed the list with favourites. The connection in this page worked perfectly from the first time and the process continued.

The “Add Task page” consists of the fields the user needs to fill in, in order to save a task with the working hours. This connection with the web service also worked well from the beginning.

Afterwards, the application was tested with the case company's representative. All the features were approved and excellent feedback was collected. In addition to that extra features were implemented and tested to two of the pages: “Add Task page” and “Favorite page”. The features were as followed:

- Calendar to both pages to choose the date to which you record hours;
- Checkbox for traveling day to both pages, to be able to mark if the hours are for travels;
- Customized messages if the hours has been saved or not;

All of the above features were tested and the end application was tested as a whole working mobile application. The feedback and the evaluation of the work is placed in section 5 Evaluation.

4.6 Implementing/ publishing and maintenance

At last the developing process ends with the Implementation and the maintenance of the finished product. As agreed at the beginning of the process the implementation and the maintenance of the hour tracking application is Evolvit Oy's responsibility. The application source code and all related material will be given to the company. The mobile device and the installed application on it will be returned to the company's representative.

Evolvit Oy will be responsible for distributing the mobile application within its employees and in the future to their clients.

5 Evaluation

The Hour tracking mobile application is currently in implementing stage. The mobile application source code and working prototype were given to Evolvit Oy. The implementation of the source code to a working application would be made by Evolvit Oy's representatives.

In the last few demonstration of the Hour tracking application the Evolvit Oy's representative, Mr.Saarikivi, was impressed of how good the application look and work. The mobile application met the customer requirements and passed successfully all the testing. After the last demonstration Mr. Saarikivi sent an email for expressing his exitment and the satisfaction.

Mr. Saarikivi is thrilled that the mobile application has all the features that were required beforehand and he especially satisfied with the layout and colours used in the mobile application. All in all Mr. Saarikivi has been very pleased with the whole development process and the delivered product.

The emails and the feedback will be attached in Appendix 3 : Customer Feedback.

6 Conclusion

The development project was successful and that mobility in Evolvit Oy will increase in the near future. The mobile application is developed using SOAP Web services, which allows the employees of Evolvit Oy, to record their work hours anytime from anywhere using their mobile device.

The Hour Tracking mobile application will save the time spent before for work recording. Before the employees needed to login through their laptops or work computers and go through several steps to record their working hours. Now using the Hour Tracking mobile application requires only login and one button pressing to record the hours into the company's system. After implementing the Hour Tracking mobile application the recording of the working hours will be more accurate. The collection of working hours can be done in the same moment when the work is finished.

However, the future favourable outcome of the mobile application obligate of the implementation within the company and possible furthermore to Evolvit Oy's clients.

Furthermore, Mr. Saarikivi has required a search function to be implemented in the application. This could be developed used the same tools and methods used in this case study: Visual Studio, SOAP, WSDL, C#, XAML. However, due to lack of time this modification, in addition to other modifications in the mobile application, could be a perfect thesis task for an Information Technology student.

References

- Adekunle, O. 2011. The constructive research approach in project management research. International Journal of Managing Projects in Business, Issue 4.
- Banfield, R., 2009. Defining and Developing a Web Prototype Design: Part 1. Accessed 3 April 2015. <http://www.freshtilledsoil.com/web-prototype-design-part-1/>
- Barry, D. K., 2013. Web Services, Service-Oriented Architectures, and Cloud Computing. Waltham: Elsevier.
- Citrix and Vanson Bourne.2013. Mobility in Business Report. Accessed 15 March 2015. https://www.citrix.com/content/dam/citrix/en_us/documents/news/citrix-mobility-in-business-report.pdf
- citrix.com, 2013. Global Survey Identifies Mobility as a Top Priority for the Enterprise. Accessed 03 February 2015. <http://www.citrix.com/news/announcements/oct-2013/global-survey-identifies-mobility-as-a-top-priority-for-the-ente.html>.
- Dawson, W. 2009. Projects in computing and information systems : a student 's guide. New York: Pearson Education.
- Egwuatu, A. 2013. RESEARCH | AGILE METHODOLOGY. Accessed 7 April 2015. <http://citewire.com/research-agile-methodology/>
- eMarketer. 2014. 2 Billion Consumers Worldwide to Get Smart(phones) by 2016. Accessed 24 May 2015. <http://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694>
- Gudgin, M. 27. SOAP Version 1.2 Part 0: Primer (Second Edition). Accessed 16 March 2016. <http://www.w3.org/TR/soap12/>
- Hejlsberg, A. Torgersen, M., Wiltamuth, S. & Golde, P., 2011. The C# Programming Language. 4th toim. Upper Saddle River, NJ: Addison-Wesley.
- Jenkov, J. 2014. Web Services Overview. Accessed 5 May 2015. <http://tutorials.jenkov.com/web-services/overview.html>
- Justin. 2013. What is the Software Development Life Cycle?. Accessed 21 April 2015. <https://airbrake.io/blog/insight/what-is-the-software-development-life-cycle>
- Kerner, S. 2014. Growing Importance of Mobile Devices in Business [Infographic]. Accessed 15 May 2015. <http://www.cpcstrategy.com/blog/2014/10/mobile-devices-infographic/>.
- Kumar, V. 2012. 101 Design Methods : A Structured Approach for Driving Innovation in Your Organization. s.l.:Wiley.
- McWherter, J. 2012. Professional Mobile Application Development. Hoboken: Wiley.
- Microsoft. 2015. ASP.NET Cookies Overview. Accessed 18 May 2015. <https://msdn.microsoft.com/en-us/library/ms178194.aspx>
- Research Methods: Summary. No Date. Accessed 20 April 2015. <http://betterthesis.dk/research-methods/research-methods-summary>
- Rice, K. 2013. Accessed 03 February 2015. <http://www.kinvey.com/blog/3339/14-surprising-stats-about-enterprise-mobility>

- Rouse, M. No Date. TechTarget. Accessed 10 My 2015.
<http://searchmobilecomputing.techtarget.com/definition/enterprise-mobility>
- Saarikivi, A. 2015. Company Requirements. Interview. 10 February 2015.
- Snell, J. & Tidwell, D. & kulchenko, P. 2001. Programming Web services with SOAP. O'Reilly Media, Inc.
- Stickdorn, M. 2012. This is Service Design Thinking : Basics-Tools-Cases. BIS Publishers.
- Stroud, D. No date . Understanding the Purpose and Use of Benchmarking. Accessed 12 March 2015. <http://www.isixsigma.com/methodology/benchmarking/understanding-purpose-and-use-benchmarking/>
- Tutorialspoint. 2014. SDLC Overview. Accessed 26 May 2015.
http://www.tutorialspoint.com/sdlc/sdlc_overview.htm .
- TSheets. 2015. Accessed 21 April 2015. <https://www.tsheets.com/> .
- Visual Studio. 2015. Accessed 12 January 2015. <http://www.visualstudio.com/> .
- W3C. 2015. Accessed 20 March 2015. <http://www.w3.org/> .
- W3C. 2015. STANDARDS. Accessed 18 March 2015. <http://www.w3.org/standards/>
- W3C. 2001. Web Services Description Language (WSDL) 1.1. Accessed 19 March 2015.
<http://www.w3.org/TR/wsdl>
- W3C. 2013. Extensible Markup Language (XML) 1.0 (Fifth Edition). Accessed 10 April 2015.
<http://www.w3.org/TR/REC-xml/>
- Wang, P. S., 2013. Dynamic Web programming and HTML5. Boca Raton: CRC Press.
- Wikipedia, 2014. Website wireframe. Accessed 10 February 2015.
http://en.wikipedia.org/wiki/Website_wireframe
- Wikipedia, 2015. C Sharp (programming language). Accessed 20 February 2015
[https://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)](https://en.wikipedia.org/wiki/C_Sharp_(programming_language))
- Wikipedia, 2015. Extensible Application Markup Language. Accessed 16 February 2015.
https://en.wikipedia.org/wiki/Extensible_Application_Markup_Language
- Wikipedia, 2015. Mobile app. Accessed 20 January 2015.
https://en.wikipedia.org/wiki/Mobile_app
- Wikipedia, 2015. SOAP. Accessed 23 January 2015. <https://en.wikipedia.org/wiki/SOAP>
- Wikipedia, 2015. Web service. Accessed 21 January 2015.
https://en.wikipedia.org/wiki/Web_service

Figures

Figure 1 The business benefits of being a mobile organization.....	7
Figure 2 Software Development Life Cycle	11
Figure 3 Agile Model.....	11
Figure 4 Web service architecture	15
Figure 5 Web service accessing model	16
Figure 6 SOAP Message Structure.....	18
Figure 7 Visual Studio Community 2013	21
Figure 8 TSheets Mobile Application	24
Figure 9 Wireframe	25
Figure 10 Development process	26
Figure 11 Hour Tracking Application v1.0.....	27
Figure 12 Hour Tracking Application v.2.0.....	27
Figure 13 Hour Tracking Application Final Version	28
Figure 14 First Testing Error Message.....	29

Tables

Table 1 WSDL Elements	19
-----------------------------	----

Appendices

Appendix 1: Customer Requirement Questionnaires	38
Appendix 2: Snapshot of Company Requirement.....	39
Appendix 3: Customer Feedback	40
Appendix 4: Prototype	41

Appendix 1: Customer Requirement Questionnaires

Interviewer: [Elena Mihaleva](#)

Interviewee: [Antti Saarikivi, Evolvit Oy](#)

Topic: [Customer Requirements](#)

Date: [11.02.2015](#)

Questions:

1. What is the main focus group for the application, should they be taken into consideration during the development?

Answer: Mainly for Evolvit staff. In future can be offered for company clients.

2. What is the main purpose of the application?

Answer: To offer better service for employees/ clients and mobile option to the existing service.

3. What key features of the application?

Answer: To enter and edit hour tracking info; Favourite list; Entering the travel information; (Advanced (GPS)).

4. What is the most important things to take into consideration during the development?

Answer: User friendliness.

5. Which operating systems (OS) to concentrate the most on?

Answer: WP8

6. Do you have any other wishes / ideas?

Answer: No.

7. Who will take care of publishing and maintenance of the mobile application?

Answer: Evolvit Oy.

8. What colour scheme to be used for the design of the mobile application?

Answer: Follow the sites colour

Appendix 2: Snapshot of Company Requirement

 <p>Hour tracking mobile app</p> <p>8.1.2015</p>	<p>Muokkaa perustyyli. napsauta </p> <ul style="list-style-type: none">• <u>Evolvit Oy</u>• <u>~70 employees</u>• <u>Espoo, Jyväskylä, Tampere</u>
<p>Project description </p> <ul style="list-style-type: none">• Hour tracking mobile app<ul style="list-style-type: none">– Save hour tracking information to our existing system– COM component between SQL database and the mobile app (C#,IIS)• Features: entering hours and description, favorite list, entering travel information (GPS)...• User experience(UX)• Platform independent (Android / iOS / WP)	<p>What we expect </p> <p><u>Required from students</u></p> <ul style="list-style-type: none">• <u>Will to learn, happy attitude & eagerness</u>• <u>Web-programming</u>• Server-side <u>programming</u>• <u>Usability & User Experince (UX) in mind</u>• <u>PhoneGap (html,css,js)</u>, AJAX, JSON, REST, .C#,SQL,IIS
<p><u>What you get</u> </p> <ul style="list-style-type: none">• A <u>real life project</u>• <u>Something for the CV</u>• <u>Possibility to get hired</u>• <u>Something for you effort after the project</u>	 <p>Yhdessä eteenpäin</p> <p>Antti Saarikivi IT-Consultant</p> <p>antti.sarikivi@evolvit.fi 040 519 5354</p> <p>www.evolvit.fi</p>

Appendix 3 : Customer Feedback

The customer has been asking for feedback during the whole development process. The feedback collected was used for constant improvement of the mobile application. After the first demonstration of the working mobile application on 28 April, 2015 Mr. Saarikivi replied, "Nice to see your work. I think this will be a successful project."

"

On 28 April 2015 at 11:36, Antti Saarikivi <Antti.Saarikivi@evolvit.fi> wrote:

Hello Elena,

Nice to see your work. I think this will be a successful project.

Mikko has added new features for the wsdl.

- External ID added for the SaveTaskLogEntry

Some new functions to get customers/projects

[...]

Check those out and let me know if you have some questions.

Happy Holiday!

Antti

"

On 3 December, 2015 the customer gave a last overall feedback of the project work.

"

On 3 December 2015 at 10:28, Antti Saarikivi <Antti.Saarikivi@evolvit.fi> wrote:

Hi Elena,

Here is your feedback. [..]

Elena was very self-imposed with this project. I had only give her a small definition about what is required and she did all the work with minimum guidance needed. Although the app was not completely finished we got nice base what we can use inhouse to develop the App to production. Elena's attitude was very good and the quality of the produced code was high.

Br,

Antti Saarikivi

"

Appendix 4: Prototype



Prototype1

09/02/15 11:15

Author: Elena Mihaleva

File name: Prototype1.vp

Version 1.0

02. Screens / Prototype1

main

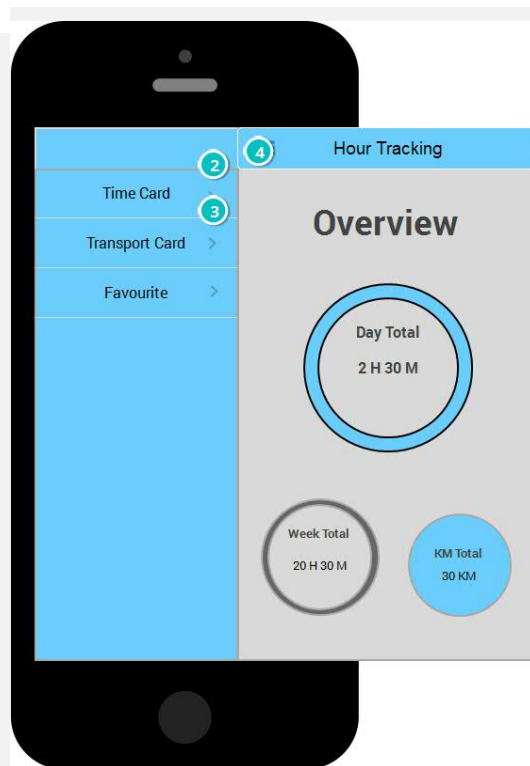


Interactions

1 on Click: goes to Screen 1 →

02. Screens / Prototype1

Screen 1



Interactions

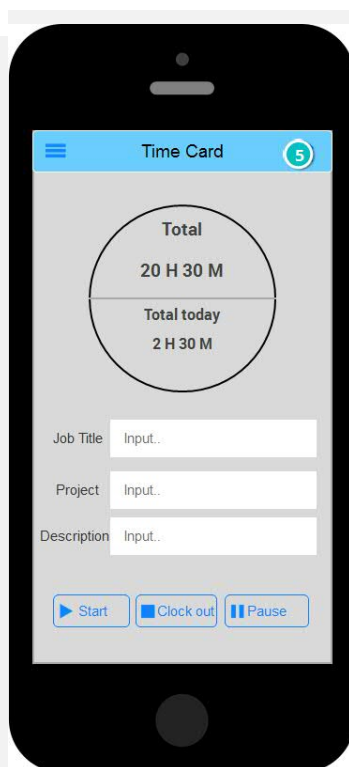
2 on Click: goes to Timecard →

4 on Click: goes to main →

3 on Click: goes to Transport →

02. Screens / Prototype1

Timecard

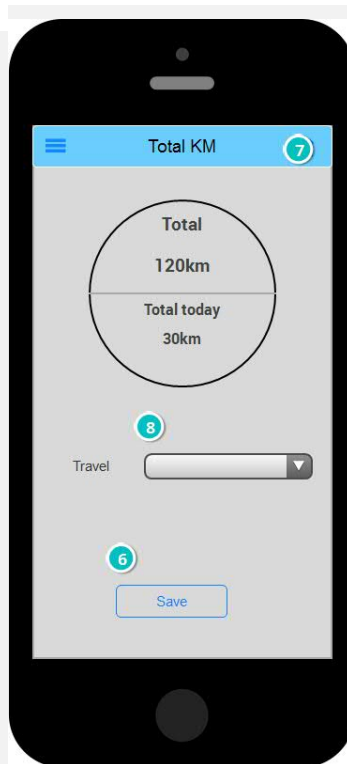


Interactions

5 on Click: goes to main →

02. Screens / Prototype1

Total Km



Interactions

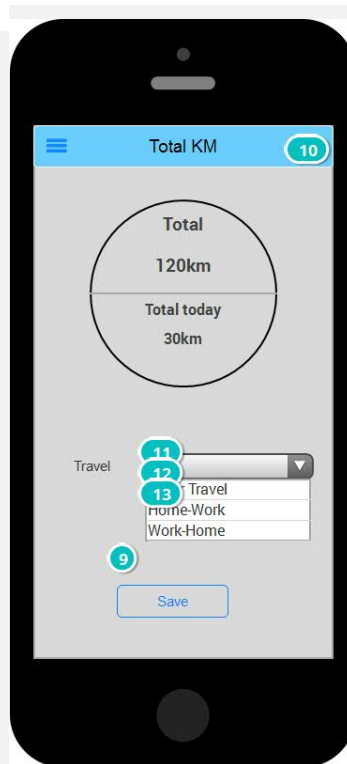
6 on Click: goes to main →

8 on Click: goes to totalkmdropdown →

7 on Click: goes to main →

02. Screens / Prototype1

totalkmdropdown



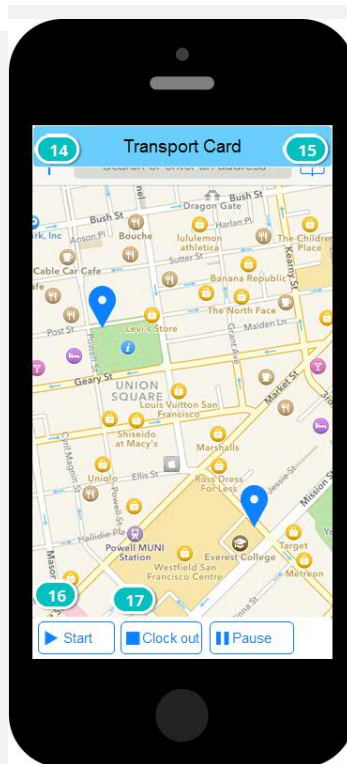
Interactions

- 9 on Click: goes to main →
- 10 on Click: goes to main →
- 11 on Click: goes to Total Km →

- 12 on Click: goes to Total Km →
- 13 on Click: goes to Total Km →

02. Screens / Prototype1

Transport



Interactions

14 on Click: goes to TransportMenu →

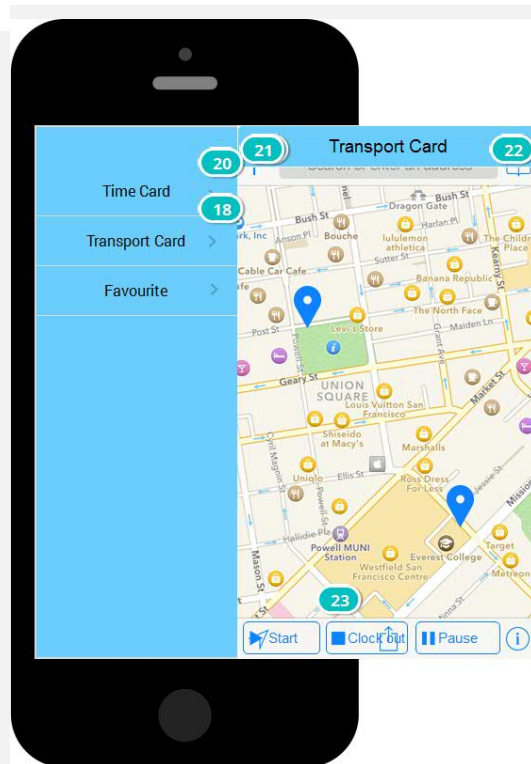
16 on Click: goes to Transportstart →

15 on Click: goes to main →

17 on Click: goes to Total Km →

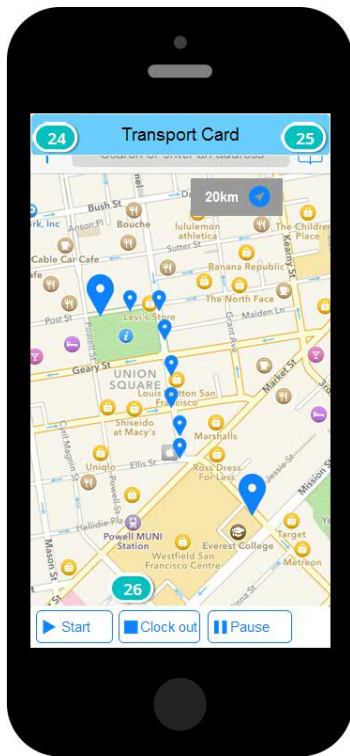
02. Screens / Prototype1

TransportMenu



Interactions

- | | |
|----------------------------------|----------------------------------|
| 18 on Click: goes to Transport → | 21 on Click: goes to Transport → |
| 19 on Click: goes to main → | 22 on Click: goes to main → |
| 20 on Click: goes to Timecard → | 23 on Click: goes to Total Km → |



Interactions

24 on Click: goes to TransportMenu →

25 on Click: goes to main →

26 on Click: goes to Total Km →